

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-69376

(43) 公開日 平成10年(1998) 3月10日

(51)Int.Cl. ⁸		識別記号	庁内整理番号	F I		技術表示箇所
G 0 6 F	9/06	4 1 0		G 0 6 F	9/06	4 1 0 S
	3/14	3 7 0			3/14	3 7 0 A
	9/445				12/00	5 4 7 H
	12/00	5 4 7			13/00	3 5 4 D
	13/00	3 5 4			9/06	4 2 0 J
審査請求 未請求 請求項の数13 O L (全 17 頁)						

審査請求 未請求 請求項の数13 O L (全 17 頁)

(21) 出願番号 特願平9-71119

(22) 出願日 平成9年(1997) 3月25日

(31) 優先権主張番号 08/621578

(32) 優先日 1996年3月26日

(33) 優先権主張国 米国 (U S)

(71) 出願人 594170738

サン マイクロシステムズ インコーポレ
イテッドアメリカ合衆国 カリフォルニア州
94043 マウンテン ヴィュー ガルシア
アヴェニュー 2550

(72) 発明者 ロバート ビー ジャーヴィス

アメリカ合衆国 カリフォルニア州
95030 モンテ セレノ ヴィア セレノ
17645

(74) 代理人 弁理士 中村 稔 (外6名)

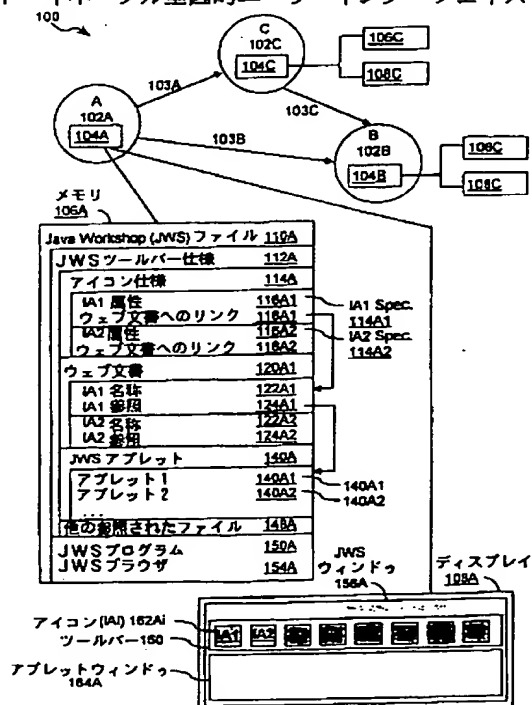
最終頁に続く

(54) 【発明の名称】 ワールド・ワイド・ウェブ文書に対する組込型リンクを有しているツールバー・アイコン及び統合型ウェブ・ブラウザを有するインターネット・イネーブル型図的ユーザ・インターフェイス

(57) 【要約】 (修正有)

【課題】 局所または遠隔ファイル及びプログラムに関連付けることができるアイコンを供給するシステム及び方法を提供する。

【解決手段】 ウェブ文書又は参照されたオブジェクトは、遠隔又は局所 (GUIを表示しているコンピュータのOSを介して直接アクセス可能) でありうる。GUIは、ツールバー・アイコンが選択されるときにはいつでもインターフェイスが選択されたアイコンにリンクされたウェブ文書をロードすべくウェブ・ブラウザをトリガするように、ウェブ・ブラウザ・ソフトウェアにリンクされ、ウェブ・ブラウザは、ウェブ文書によって参照されたファイルを自動的にロードし実行可能なものを実行する。各ツールバー・アイコンは、アイコンの機能をインプリメントする単一アプレット (遠隔または局所) にリンクされる。アイコンの関連ウェブ文書とウェブ・ブラウザの間の結合により、関連アイコンは、アイコンが選択されるときは自動的に実行される。



【特許請求の範囲】

【請求項1】 ディスプレイ及びメモリを有している第1のコンピュータにおいて、前記第1のコンピュータは、一組の遠隔コンピュータでネットワークされており、そのコンポーネントが前記コンピュータ上でだけ存在すべく拘束されていない複合文書のローディング及び実行を前記第1のコンピュータ上に表示された図的ユーザ・インターフェイスから起動するためのシステムであって：一組の選択可能なアイコンを有しており、各前記アイコンが当該アイコンの選択を介して起動される一つ以上の操作に関連付けられている、表示可能ツールバー；それぞれが前記関連付けられた操作の一つ以上を実行するために必要なコンポーネント、またはファイルに対する一組の参照を含む、一組の複合文書；アプレットまたはデータ・ファイルから選択されたファイル型及び局所または遠隔から選択された位置を有している前記ファイルのサブセットのそれぞれ；それぞれが前記アイコンの一つを前記文書の組の一つ以上に関連付け、かつそれぞれが前記文書が前記コンピュータ上に記憶されるときに局所リンクからまたは前記文書が前記遠隔コンピュータ上に記憶されるときにネットワーク・リンクから選択される、一組のリンク；及び前記アイコンの一つが選択されるときに、前記コンピュータに、前記一つ組のリンクを介して前記選択されたアイコンに関連付けられた前記文書において参照された前記ファイルをロードさせかつ実行可能である前記ロードされたファイルのいずれかを実行させるように構成され、それによって前記アイコンに関連付けられた操作を起動する、ブラウザを備えていることを特徴とするシステム。

【請求項2】 前記アプレットは、前記ブラウザによってインプリメントされるバーチャル・マシンで解釈されかつ実行されるプラットフォーム独立コンピュータ言語で書かれることを特徴とする請求項1に記載のシステム。

【請求項3】 前記プラットフォーム独立コンピュータ言語は、Javaであることを特徴とする請求項2に記載のシステム。

【請求項4】 前記複合文書は、HTML文書であり、前記参照のそれぞれは、前記参照されたファイルの位置が局所であるときにファイル名から、または前記参照されたファイルの位置が遠隔であるときにURLから選択されることを特徴とする請求項3に記載のシステム。

【請求項5】 前記HTML文書にリンクされた前記アイコンが選択されるときに、前記ブラウザが前記アプレットをロードしかつ実行し、それによって、前記選択されたアイコンに関連付けられた全ての操作及びユーザ及びシステム対話を調整しかつ制御するように；前記HTML文書は、前記アイコンに関連付けられる前記操作の全てを調整するアプレットに対する単一参照を含むことを特徴とする請求項4に記載のシステム。

【請求項6】 一つのアプレットに関連付けられたアイコン仕様ファイルを更に備え、前記アプレット仕様ファイルは、その関連付けられたアプレットがその上で操作されるべき一組の入力ファイルを示しており、前記入力ファイルは、前記第1のコンピュータ上にのみ位置決めされるべく拘束されていないことを特徴とする請求項5に記載のシステム。

【請求項7】 前記第1のコンピュータに遠隔的に記憶された前記入力ファイルのそれぞれは、前記アプレット仕様ファイルのURLによって参照されかつ前記第1のコンピュータに記憶された前記入力ファイルのそれぞれは、局所経路及びファイル名によって参照されることを特徴とする請求項6に記載のシステム。

【請求項8】 コンポーネントのサブセットをホストすることができる一組の遠隔コンピュータとネットワークされている第1のコンピュータ上にだけその該コンポーネントが存在すべく拘束されていない複合文書のローディング及び実行を前記第1のコンピュータ上に表示された図的ユーザ・インターフェイスから起動する方法であって：一組の選択可能なアイコンを有しているツールバーを表示し、

各前記アイコンは、当該アイコンの選択を介して起動される一つ以上の操作に関連付けられかつ一つ以上の複合文書にリンクされ、それぞれは、前記関連操作の一つ以上を実行するために必要なコンポーネント、またはファイルに対する一組の参照を含む、前記ファイルのサブセットのそれぞれは、アプレットまたはデータ・ファイルから選択されたファイル型を有しており；前記アイコンの一つが選択されることにより、前記選択されたアイコンにリンクされた前記一つ以上の複合文書をローディングし；前記一つ以上の複合文章をローディングすることにより、前記複合文書で参照された前記ファイルをローディングし、

前記ファイルは、前記第1のコンピュータ上にだけ存在すべく拘束されておらず；かつロードされたファイルがアプレットまたはスタンドアロン・アプリケーションから選択される型を有するとき、前記ロードされたファイルを実行する段階を具備することを特徴とする方法。

【請求項9】 前記アプレットは、前記ブラウザによってインプリメントされるバーチャル・マシンで解釈されかつ実行されるプラットフォーム独立コンピュータ言語で書かれることを特徴とする請求項8に記載の方法。

【請求項10】 前記一つ以上の複合文書をローディングする段階は、：ロードされるべき複合文書が前記第1のコンピュータに記憶されるとき、前記リンクの前記複合文書に対して供給された局所経路及びファイル名を用いて前記複合文書を検索し、；かつロードされるべき前記複合文書が前記遠隔コンピュータの一つに記憶されるときに、前記リンクの前記複合文書に対して供給されたネットワーク・ノード及びファイル名を用いて前記ネッ

トワークにわたり前記複合文書に対する文書要求メッセージを発行する段階を具備することを特徴とする請求項8に記載の方法。

【請求項11】 前記ファイルをローディングする段階は、：参照されたファイルが前記第1のコンピュータに記憶されるとき、前記参照の前記参照されたファイルに対して供給された局所経路及びファイル名を用いて前記参照されたファイルを検索し、；かつ参照されたファイルが前記遠隔コンピュータの一つに記憶されるときに、前記参照の前記参照されたファイルに対して供給されたネットワーク・ノード及びファイル名を用いて前記ネットワークにわたり前記参照されたファイルに対する文書要求メッセージを発行する段階を具備することを特徴とする請求項10に記載の方法。

【請求項12】 前記複合文書は、HTML文書であり、前記参照のそれぞれは、前記参照されたファイルが前記第1のコンピュータに記憶されるときに局所参照から選択され、または前記参照されたファイルが前記一組の遠隔コンピュータの一つに記憶されるときに汎用資源ロケータ（URL）から選択されることを特徴とする請求項11に記載の方法。

【請求項13】 前記HTML文書は、前記アイコンに関連付けられる前記操作の全てを調整するアプレットに対する単一参照を含み、；かつローディングしかつ実行する前記段階は、前記アプレットをローディングしかつ実行する段階を具備し、それによって前記選択されたアイコンに関連付けられた全ての操作及びユーザ及びシステム対話を調整しかつ制御することを特徴とする請求項12に記載の方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、一般に図的ユーザ・インターフェイスに関し、特に、それらのリンクされたアイコンが選択されるときにロードされかつ実行される対応ウェブ文書にツールバーのアイコンがリンクされる図的ユーザ・インターフェイスの設計に関する。

【0002】

【従来の技術】図的ユーザ・インターフェイス（しばしばGUIsと称される）は、それによってユーザがコンピュータ・プログラム及びファイルと対話することができるよく知られた機構である。一般的なGUIは、それぞれが、GUIを制御しているプログラによって供給される特定の操作または制御しているプログラムからアクセスすることができるファイルに関連付けられた、一組の選択可能なアイコンを供給する。GUIからの操作を起動することを欲しているユーザは、適切なアイコンを（例えば、マウスで）選択することによってそのようにする。例えば、ワードプロセッサのユーザは、ワードプロセッサのツールバーからスペルチェック・アイコンを選択することによってアクティブ文書上でスペルチ

ェッキングを起動することができる。同様に、ユーザは、ファイルのアイコンを選択し、宛先ディレクトリを表している別のアイコンに選択したアイコンをドラッグし、次いでアイコンをドロップする（選択解除する）ことによって一つのディレクトリから別のディレクトリへファイルを移動することができる。この型のGUI設計は、また、（例えば、実行可能なプログラムを表しているアイコン上でダブルクリックすることによって）スタンドアロン・プログラムをランチするために適切である。

【0003】これらの場合のそれぞれにおいて、アイコンの選択によって起動された即時操作は、選択されたアイコンを表示したオペレーティング・システムまたはアプリケーション・プログラムをホストしているネットワークまたはコンピュータに対して局所（ローカル）でなければならぬ。これは、例えば、遠隔システムに配置された実行可能プログラムにアイコンを直接関連付けることができないことを意味する。その代わり、従来技術では、遠隔プログラムの実行は、遠隔システムとの通信を確立する局所実行可能プログラムに遠隔プログラム用のアイコンをリンクし、次いで局所システムに遠隔プログラムをダウンロードして実行することによって、間接的にだけ進めることができる。もちろん、このスキームは、局所プログラム及び遠隔システムがコンパチブルな通信モードを支持しかつ（それを）局所システムで実行することができるように遠隔プログラムが予めコンパイルされるときにだけ動作する。

【0004】従って、オペレーティング・システムまたはアイコンを表示しているアプリケーションのサービスを用いて遠隔オブジェクトをダウンロード及び／又は実行することができるように、プログラムまたはファイルのような、遠隔オブジェクトにアイコンをリンクさせる図的ユーザ・インターフェイスの必要性が存在する。また、アイコン選択事象を処理する局所サービス及び遠隔オブジェクトがその上に記憶される遠隔システムが通信できることを保証するホスト・システムに関連付けられるようなGUIの必要性も存在する。更に、全ての実行可能な、遠隔ファイルのサブセットは、遠隔システムが通信できることを保証されるあらゆるシステム上で実行可能であることが望ましい。これらの特徴のあるものは、それぞれが固有の汎用資源ロケータ（URLs）によって識別された文書を記憶している、インターネットを構成しているサーバの多くをリンクする、ワールド・ワイド・ウェブ（“WWW”）において実施される。ウェブ・サーバに記憶された文書の置くは、HTML（ハイパーテキスト・マークアップ言語）と呼ばれる標準文書記述言語で書かれている。HTMLを用いて、ウェブ文書の設計者は、ハイパーテキスト・リンクまたはアンカーを文書中の特定の語または語句に関連付け（これらのハイパーテキスト・リンクは、語または語句

に関する情報を供給することにより他のウェブ文書のURLsまたは同じ文書の他の部分を識別する)かつ視覚的態様及びウェブ・ページの内容を特定することができる。

【0005】ユーザは、インターネットに接続されたウェブ・クライアント上で走っているウェブ・ブラウザ(HTML文書を表示しかつウェブ・サーバと通信するために設計されたコンピュータ・プログラム)を用いてWWWに記憶される文書をアクセスする。一般的に、これは、ウェブ・ブラウザで見られている文書内のハイパーテキスト・リンク(強調された語または語句としてウェブ・ブラウザによって一般的に表示される)を選択しているユーザによって行われる。ウェブ・ブラウザは、次いで、要求した文書のURLによって識別されたウェブ・サーバに、要求した文書に対するHTTP(ハイパーテキスト・トランスファ・プロトコル)要求を発行する。応答として、指定されたウェブ・サーバは、HTTPをまた用いて、ウェブ・ブラウザに要求した文書をリターンする。

【0006】ウェブ・ページの標準HTML構文及びWWWによって支持される標準通信プロトコル(HTTP)は、あらゆるウェブ・ブラウザがあらゆるウェブ・サーバと通信することができることを保証する。しかしながら、Javaプログラミング言語及びJavaアプレットの発明まで、インターネット及びWWWにわたりプラットフォーム-独立アプリケーション・プログラムを供給する方法が存在しなかった。

【0007】Javaプログラミング言語の重要な特徴は、Javaインタプリタを有しているあらゆるコンピュータ・プラットフォーム上でそれらを実行することができるということを意味する、Java言語で書かれたプログラムのアーキテクチャ-独立、及び、Javaプログラムの保天性(完全性)をそれらの実行の前に検証することができるということを意味する、そのようなプログラムの保天性の検証可能性を含む。Javaプログラム・ベリファイヤは、プログラムが、検証されたプログラムが実行しているコンピュータのオペランド・スタックをオーバーフローまたはアンダーフローできないということを確実にしかつ全てのプログラム命令が既知のデータ・タイプのデータだけを利用するということを確実にする所定のスタック用途及びデータ用途制限に準拠するか否かを決定する。結果として、Java言語プログラムは、オブジェクト・ポインタを生成することができないしかつ一般に、ユーザが使用するためのの許可をそれに明示的に認可するそれらの資源以外のシステム資源をアクセスすることができない。結果として、一つ以上のコード・フラグメントが関連フォームまたはイメージ・ファイルと一緒にクライアントにダウンロードされる

ムに記入することが必要なダウンロードされたコード・フラグメントを検証しかつ実行することができる。

【0008】

【発明が解決しようとする課題】それゆえに、Java-コンパチブル・ウェブ・ブラウザは、局所(ローカル)システム上で実行可能であることを保証されるJavaアプレットをあらゆるWWWサーバからダウンロードすることができる。しかしながら、既存のウェブ・ブラウザは、ブラウザによって現在表示されているウェブ・ページにおいてハイパーリンクを選択しているユーザ、またはブラウザの中にURLを入力しているユーザに応じてウェブ文書だけをダウンロードするように構成されている。ウェブ・ブラウザは、遠隔オブジェクト、実行可能なプログラム(スタンドアロン・プログラムまたはアプレット)または他のファイルのいずれか、に関連付けられるアイコンをGUIから選択しているユーザに応じてこれらのウェブに関するサービスを供給するように構成されていない。それゆえに、局所(ローカル)または遠隔ファイル及びプログラムに関連付けることができるアイコンを供給するGUIの必要性が存在する。次いで、遠隔ファイルまたはプログラムに関連付けられたアイコンが選択されるとき、このGUIは、遠隔ファイルまたはプログラムのURL上の局所ウェブ・ブラウザ・サービスを呼出すべきであり、可能であれば、局所マシンにダウンロードされかつ実行されるファイルまたはプログラム、及びそのようなファイルまたはプログラムにおいて参照された全ての他のオブジェクトを結果として生ずる。

【0009】本発明の目的は、上記従来の技術における問題点を鑑み、局所または遠隔ファイル及びプログラムに関連付けることができるアイコンを供給することができるシステム及び方法を提供することである。

【0010】

【課題を解決するための手段】本発明の上記目的は、ディスプレイ及びメモリを有している第1のコンピュータにおいて、第1のコンピュータは、一組の遠隔コンピュータでネットワークされており、そのコンポーネントがコンピュータ上でだけ存在すべく拘束されていない複合文書のローディング及び実行を第1のコンピュータ上に表示された図的ユーザ・インターフェイスから起動するためのシステムであって：一組の選択可能なアイコンを有しており、各アイコンが当該アイコンの選択を介して起動される一つ以上の操作に関連付けられている、表示可能ツールバー；それぞれが関連付けられた操作の一つ以上を実行するために必要なコンポーネント、またはファイルに対する一組の参照を含む、一組の複合文書；アプレットまたはデータ・ファイルから選択されたファイル型及び局所または遠隔から選択された位置を有しているファイルのサブセットのそれぞれ；それぞれがアイコンの一つを文書の組の一つ以上に関連付け、かつそれぞ

れが文書がコンピュータ上に記憶されるときに局所リンクからまたは文書が遠隔コンピュータ上に記憶されるときにネットワーク・リンクから選択される、一組のリンク：及びアイコンの一つが選択されるときに、コンピュータに、一組のリンクを介して選択されたアイコンに関連付けられた文書において参照されたファイルをロードさせかつ実行可能であるロードされたファイルのいずれかを実行させるように構成され、それによってアイコンに関連付けられた操作を起動する、ブラウザを備えているシステムによって達成される。

【0011】本発明のシステムでは、アプレットは、ブラウザによってインプリメントされるバーチャル・マシンで解釈されかつ実行されるプラットフォーム独立コンピュータ言語で書かれてもよい。本発明のシステムでは、プラットフォーム独立コンピュータ言語は、Javaであるように構成してもよい。本発明のシステムでは、複合文書は、HTML文書であり、参照のそれぞれは、参照されたファイルの位置が局所であるときにファイル名から、または参照されたファイルの位置が遠隔であるときにURLから選択されるように構成してもよい。本発明のシステムでは、HTML文書にリンクされたアイコンが選択されるときに、ブラウザがアプレットをロードしかつ実行し、それによって、選択されたアイコンに関連付けられた全ての操作及びユーザ及びシステム対話を調整しかつ制御するように；HTML文書は、アイコンに関連付けられる操作の全てを調整するアプレットに対する単一参照を含むように構成してもよい。

【0012】本発明のシステムでは、一つのアプレットに関連付けられたアイコン仕様ファイルを更に備え、アプレット仕様ファイルは、その関連付けられたアプレットがその上で操作されるべき一組の入力ファイルを示しており、入力ファイルは、第1のコンピュータ上にのみ位置決めされるべく拘束されていないように構成してもよい。

【0013】本発明のシステムでは、第1のコンピュータに遠隔的に記憶された入力ファイルのそれぞれは、アプレット仕様ファイルのURLによって参照されかつ第1のコンピュータに記憶された入力ファイルのそれぞれは、局所経路及びファイル名によって参照されるように構成してもよい。また、本発明の上記目的は、コンポーネントのサブセットをホストすることができる一組の遠隔コンピュータとネットワークされている第1のコンピュータ上にだけその該コンポーネントが存在すべく拘束されていない複合文書のローディング及び実行を第1のコンピュータ上に表示された図的ユーザ・インターフェイスから起動する方法であって：一組の選択可能なアイコンを有しているツールバーを表示し、各アイコンは、当該アイコンの選択を介して起動される一つ以上の操作に関連付けられかつ一つ以上の複合文書にリンクされ、それぞれは、関連操作の一つ以上を実行するために必要

なコンポーネント、またはファイルに対する一組の参照を含み、ファイルのサブセットのそれぞれは、アプレットまたはデータ・ファイルから選択されたファイル型を有しており；アイコンの一つが選択されることにより、選択されたアイコンにリンクされた一つ以上の複合文書をローディングし；一つ以上の複合文章をローディングすることにより、複合文書で参照されたファイルをローディングし、ファイルは、第1のコンピュータ上にだけ存在すべく拘束されておらず；かつロードされたファイルがアプレットまたはスタンドアロン・アプリケーションから選択される型を有するとき、ロードされたファイルを実行する段階を具備する方法によって達成される。

【0014】本発明の方法では、アプレットは、ブラウザによってインプリメントされるバーチャル・マシンで解釈されかつ実行されるプラットフォーム独立コンピュータ言語で書かれるようにしてもよい。本発明の方法では、一つ以上の複合文書をローディングする段階は、：ロードされるべき複合文書が第1のコンピュータに記憶されるとき、リンクの複合文書に対して供給された局所経路及びファイル名を用いて複合文書を検索し、；かつロードされるべき複合文書が遠隔コンピュータの一つに記憶されるときに、リンクの複合文書に対して供給されたネットワーク・ノード及びファイル名を用いてネットワークにわたり複合文書に対する文書要求メッセージを発行する段階を具備するようにしてもよい。

【0015】本発明の方法では、ファイルをローディングする段階は、：参照されたファイルが第1のコンピュータに記憶されるとき、参照の参照されたファイルに対して供給された局所経路及びファイル名を用いて参照されたファイルを検索し、；かつ参照されたファイルが遠隔コンピュータの一つに記憶されるときに、参照の参照されたファイルに対して供給されたネットワーク・ノード及びファイル名を用いてネットワークにわたり参照されたファイルに対する文書要求メッセージを発行する段階を具備するようにしてもよい。本発明の方法では、複合文書は、HTML文書であり、参照のそれぞれは、参照されたファイルが第1のコンピュータに記憶されるときに局所参照から選択され、または参照されたファイルが一組の遠隔コンピュータの一つに記憶されるときに汎用資源ロケータ（URL）から選択されるようにしてもよい。

【0016】本発明の方法では、HTML文書は、アイコンに関連付けられる操作の全てを調整するアプレットに対する単一参照を含み；かつローディングしかつ実行する段階は、アプレットをローディングしかつ実行する段階を具備し、それによって選択されたアイコンに関連付けられた全ての操作及びユーザ及びシステム対話を調整しかつ制御するようにしてもよい。

【0017】

【作用】本発明は、図的ユーザ・インターフェイスの一

組の構成要素を表示すべく一組の遠隔コンピュータでネットワークされたコンピュータを指図しかつ表示された組の構成要素の一つ以上の選択によって示された操作を起動すべく構成されたコンピュータ読取り可能メモリである。このコンピュータ・メモリは、ブラウザ・ソフトウェア、及び一組の構成要素に関連付けられた特徴を画定するツールバー仕様を含む。より特定的には、ツールバー仕様は、一組の構成要素における各構成要素に対して一組の視覚的属性及び関連ファイルに対する一組のリンクを画定する。

【0018】一組の視覚的属性は、コンピュータによって表示されたときの構成要素の外観（外形）を画定する。一組のリンクにおける各リンクは、構成要素に関連付けられた操作をインプリメントすることが必要なファイルへの参照を含む文書に構成要素に関連付ける。参照されたファイルは、アプレットまたはデータ・ファイルから選択されたタイプ（型）を有することができ、かつコンピュータまたは遠隔コンピュータの一つに配置することができる。結果として、リンクは、コンピュータに記憶されたファイルに対する局所リンクまたは遠隔コンピュータに記憶されたファイルに対するネットワーク・リンクから選択することができる。これらのデータ構造を与えて、構成要素の一つが選択されるとき、ブラウザは、一組のリンクを介して選択された構成要素に関連付けられた文書で参照されたファイルをロードしかつ実行可能であるロードされたファイルのいずれかを実行すべくコンピュータに指図する。

【0019】本発明の更なる目的及び特徴は、添付した図面を参照して以下の詳細の説明及び特許請求の範囲からより容易に明らかであろう。

【0020】

【実施例】図1を参照すると、少なくとも3つのコンピュータ：A 102A、B 102B、及びC 102Cを有するコンピュータ・ネットワーク100が示されている。各コンピュータ102は、プロセッサ104、高速、主メモリ（例えば、RAM）またはより遅い、二次メモリ（例えば、ハードディスク・ドライブ）である、メモリ106、及びディスプレイ108を含む。コンピュータ102は、よく知られた計算原理に従って操作する（即ち、各コンピュータ102は、実行しているプログラムに対するシステム・サービスを提供する、オペレーティング・システム（図示省略）の制御下でそのメモリ106のプログラムを実行する）。好ましい実施例では、本発明がネットワークされたコンピュータが（HTTPのような）標準通信プロトコルを用いて通信することができかつプラットフォーム独立プログラムがブラウザ・ソフトウェア内からネットワークにわたりダウンロードされかつ実行されることができるようなあらゆる環境にも適用可能であるが、コンピュータ102間の相互接続103は、インターネットによって供給され

る。本発明の操作を説明する目的のために、3つのコンピュータA、B及びCの間でファイル交換操作を調整するネットワーク・オペレーティング・システムが存在しないということが想定される。

【0021】図1に示された特定のインプリメンテーションに関して本発明の詳細がここで説明される。このインプリメンテーションでは、本発明のユーザ・インターフェイスは、とりわけ、ユーザにポートフォリオと呼ばれるコレクション（収集）の中に実行可能プログラム

10 （Javaアプレット及びスタンドアロン・エクスクータブル）及び非実行可能ファイル（イメージ・ファイル及びJavaクラス・ライブラリ）を編成させる、Java Workshop（JWS）プログラム150Aと呼ばれるアプリケーション内に埋込まれる。プログラム及びファイル・マネージャの領域における従来技術からの主な脱却において、JWSプログラム150Aは、遠隔（ユーザのマシンまたはローカル・ネットワークから離れて記憶された）または局所であるポートフォリオをユーザに生成させかつワークさせる統合型（集積型）JWSブラウザ154Aを有する。更に、JWSブラウザ154Aは、ポートフォリオを局所及び遠隔“プロジェクト”の混合であるようにアSEMBLさせる。

【0022】用語“プロジェクト”は、ポートフォリオのコンポーネントを意味すべくこの文書の目的に対して画定される。

【0023】本発明のユーザ・インターフェイスは、オブジェクトの位置に係わりなく全てのオブジェクトとワークするための単一の模範(single paradigm)を供給することによって、局所及び遠隔プロジェクトの両方から構成されているポートフォリオのような、混合されたオブジェクトとのユーザ対話を容易にする。勿論、遠隔オブジェクトとワークすること及び局所オブジェクトとワークすることの間に相違が存在する。例えば、遠隔コンピュータに記憶されたJavaアプレットを実行することは、局所的に記憶されたスタンドアロン・プログラムを実行することとは非常に異なるタスクである。これらの相違は、JWSプログラム150Aで処理される。しかしながら、本発明のユーザ・インターフェイスは、ユーザに同じ方法で（例えば、アプレットを表しているアイコンをダブル・クリックすることによって）遠隔アプレットまたは局所プログラムの実行を起動させる。どのように本発明のユーザ・インターフェイスは、この位置一透過可撓性を供給するのかがここで図1を参照して説明する。

【0024】図1を参照すると、メモリ106Aは、Java Workshop（JWS）を構成するユーザ・インターフェイス、方法及びデータ・ファイルを収集的に画定する一組のJWSファイル110Aを含む。より特定的に、JWSファイル110Aは、JWSプログラム150（以後“JWS”と称する）、JWSブラウ

ザ154A及び、JWSツールバー仕様112Aと呼ばれる一群のインターフェイス・ファイルを含む。JWSツールバー仕様112Aは、4つのファイルのサブグループ：アイコン仕様114A、ウェブ文書120A、JWSアプレット140A及び他の参照ファイル148Aから構成される。構成要素114A、120A、140A、148Aは、外観（外見）、かつより重要なのは、JWSツールバー160の構成要素としてディスプレイ108A上に表示される一組のアイコン（IAi）162Aiの操作を特定する。JWSユーザ・インターフェイスのキー構成要素である、JWSツールバー160は、JWSウィンドウ156A上にJWSプログラム150Aによって表示される。JWSウィンドウ156Aは、また、プロジェクト及び／又はポートフォリオ管理のうちにJWSプログラム150Aによって実行されるJWSアプレット140Aによって制御されるアプレット・ウィンドウ164Aを含む。

【0025】各アイコンIAiは、アイコンの視覚的属性116Aiを画定しかつ、アイコンIAiが選択されるときにはいつでもロードされ、あるいは実行されるべきファイルの最初の組をリストするウェブ文書120Aiへのリンク118Aiを特定する対応アイコン仕様114Aを有する。リンク118Aiは、局所システム（例えば、コンピュータ102A）に記憶されるウェブ文書120Aiに対するものであることができ、そのような場合には、リンクは、局所オペレーティング・システム（図示省略）によって供給されるファイル・サービスによって処理することができる局所経路及びファイル名を含む。リンク118Aiは、また、通常のウェブ・ブラウザによりインターネット上で検索することができる遠隔ウェブ文書（例えば、コンピュータ102B、102Cに記憶された文書）に対するものであることもできる。JWSプログラム150Aは、通常のウェブ・ブラウザの特徴の全てを供給するJWSブラウザ154Aを組み込むので、特定のアイコンIAiにリンクされたウェブ文書120Aiがどこに記憶されるかということをかまわないし、どの型のプラットフォームにリンクされた文書が記憶されるかということにもかまわない。好ましい実施例に対して重要なことは、JWSブラウザ154Aが、HTTPまたはFTPのような、インターネットによって支持された標準通信プロトコルの一つを介してウェブ文書120A1をホストしている遠隔プラットフォームと通信することができるということである。そうであるならば、リンクしたウェブ文書120Aiは、それらの対応アイコンが選択されるときにはいつでも（JWSプログラム150Aによってトリガされる）JWSブラウザ154Aによって自動的にダウンロードされる。これは、遠隔の、実行可能な文書にアイコンをリンクする同様な特徴をインプリメントするために従来技術において要求されるであろう複雑性の多くを削除す

る。

【0026】（メモリ106Aにロードされる前に局所的または遠隔的に最初に記憶されたであろう）各ウェブ文書120Aは、二つの構成要素を含む：名称（タイトル）122Ai及びそのコンポーネントに対する一組の参照124Ai。ウェブ文書120Aは、また、埋込まれたファイル（図示省略）を含むことができる；しかしながら、ウェブ・ブラウザは、埋込まれたファイルと参照されたファイルの間で機能的区別を行わないし、このアプリケーションも行わない。リンク118Aiにおけるように、ウェブ文書120Aの参照124Aiは、遠隔または局所ファイルに対するものであることができる。いずれの場合にも、それらは、リンク118Aiに対して説明したのと同じようにJWSブラウザ154Aによって処理される。本発明のユーザ・インターフェイスの一つの重要な利点は、参照124Aiが、その関連ウェブ文書がアプレット140Aiを参照したアイコンIAiに関連付けられた操作を処理する原因であるJavaアプレット140Aiに対するものであることができるということである。この状況において、JWSブラウザ154Aが選択されたアイコンIAiにリンクされたウェブ文書120Aiを検索するときに、それは、自動的にプル・インしかつ（遠隔システムに記憶された）参照されたアプレット140Aを実行し始める。JWSブラウザのバーチャル・マシンで走行している、アプレット140Aiは、次いで、局所オペレーティング・システム及びJWSブラウザ154Aによってそれぞれ処理される、ネットワーク及びオペレーティング・システムの複雑性について心配することを必要とせずにアイコンの関連操作をインプリメントすることができる。

【0027】好ましい実施例では、単一のJWSアプレット140Aiは、各ウェブ文書120Aiで参照される。この単一アプレットは、一つのアイコンIAiに関連付けられた機能の全てを制御するかまたは直接的にインプリメントする。例えば、好ましい実施例では、スペル・チェッカー・アイコンIA1は、遠隔アプレット140A1にウェブ文書120A1を介してリンクされうるし、一度コンピュータ102AにダウンロードされかつJWSブラウザ154Aによって実行されたならば、適切な文書をスペル・チェックする。代替的に、ウェブ文書120Aiは、多くのアプレット140Aiを参照することができる。例えば、アイコンIA2は、アイコンIA2がツールバー160から選択されるときにはいつでも両方がJWSブラウザ154Aによって自動的に送り出されるようにスペル・チェッカー・アプレット及びグラマー・チェッカー・アプレットを参照するウェブ文書120A2にリンクされうる。アプレットに加えて、本発明のウェブ文書は、データ及びイメージ・ファイルを含んでいる、他の型のコンポーネント148Aを参照することができる。

【0028】図2を参照すると、ツールバー160から特定のアイコンIA1の選択に応じてアプレットが呼出される一連の段階を説明しているデータ・フロー図が示されている。各アイコン選択事象は、アイコンIA1の選択に続いて、アイコンIA1の仕様ファイル114A1からリンク118A1を検索する、JWSブラウザ154Aによって監視される。リンク118A1を介して、アイコンIA1は、JWSブラウザ154Aによって自動的にロードされる、ウェブ文書120A1に関連付けられる。JWSブラウザ154Aは、次いで、文書120A1において参照されたファイルの全てをロードしかつまた実行可能である参照されたファイル（即ち、アプレット）を実行する。この例では、一つの参照された実行可能な、アプレット140A1が存在するということが想定される。一度それがアクティブであれば、アプレット140A1は、その上に結果、アプレットの機能及びケイパビリティとのユーザ対話を容易にするダイアログ・ボックス及びアイコンを表示することができる、ディスプレイ108Aの一部（例えば、アプレット・ウィンドウ164A）の制御を行うことができる。

【0029】図3を参照すると、Java Workshop Program（ジャワ・ワークショップ・プログラム）150Aによって用いられるメモリ106Aに記憶されるデータ項目のさらなる詳細を並べているデータ構造図が示されている。これらのデータ項目は、ツールバー160上にJWS150Aによって表示されたアイコンIAiに対する視覚的属性116Ai及びウェブ文書リンク118Aiを画定するアイコン仕様114Aiを含む。これらのアイコン（図2に示す）は、ポートフォリオ・マネージャ・アイコンIA1、プロジェクト・マネージャ・アイコンIA2、テキスト・エディット・プロジェクト・アイコンIA3、ビルド／コンパイル・アイコンIA4、ソース・ブラウザ・アイコンIA5、デバッグ・アイコンIA6、ラン・アイコンIA7及びヘルプ・アイコンIA8を含む。選択されたとき、アイコンIA1～IA8は、それぞれユーザにJWSポートフォリオ及びプロジェクトを扱わせるためにJava Workshop150Aによって供給された次に示すケイパビリティ（及びメニュー）をアクセスさせる：

【0030】IA1： 現行ポートフォリオのプロジェクトを表示する、ポートフォリオ・マネージャ・アプレット140A1へのアクセスを供給する；

IA2： ユーザがプロジェクト情報を編集することができるような、プロジェクト・マネージャ・アプレット140A2へのアクセスを供給する；

IA3： ユーザがプロジェクト・ソース・コードを編集することができるようなJWSテキスト・エディタ・アプレット（図示省略）へのアクセスを供給する；

IA4： JWSプロジェクト・コンパイラ・アプレッ

ト（図示省略）へのアクセスを供給する；

IA5： ユーザに他の参照されたファイル148Aに含まれたJWSソース・プログラムをブラウズさせるJWSブラウザ・アプレット（図示省略）へのアクセスを供給する；

IA6： ユーザに他の参照されたファイル148Aに含まれたJWSソース・プログラムをデバッグさせるJWSデバッグ・アプレット（図示省略）へのアクセスを供給する；

IA7： 実行可能なプロジェクト（即ち、アプレット及びスタンドアロン・プログラム）を走らせるJWSプロジェクト走行方法146A2fへのアクセスを供給する；及び

IA8： JWS操作に対する文脈依存ヘルプを供給するJWSヘルプ・アプレット（図示省略）へのアクセスを供給する。

【0031】図1を参照して説明されるように、好ましい実施例では、アイコン仕様114Aiは、対応しているアイコンIAiに関連付けられた操作をインプリメントする単一アプレット140Aiへの参照124Aiを有するウェブ文書120Aiへのリンクを含む。それゆえに、ポートフォリオ・マネージャ・アイコンIAiに関連付けられる、アイコン仕様114Aiは、ポートフォリオ・マネージャ・アプレット140A1に対する単一参照118Aを含むウェブ文書（“Portfolio.HTM”）120A1にリンクされる。同様に、プロジェクト・マネージャ・アイコンIA2に関連付けられる、アイコン仕様114A2は、プロジェクト・マネージャ・アプレット140A2に対する単一参照118A2を有しているウェブ文書（“Portfolio.HTM”）120A2にリンクされる。これらのアプレット140A1、140A2は、ポートフォリオ及びプロジェクトにそれぞれ適用することができる方法146A1、146A2を供給する。

【0032】方法146Aiは、それらの関連アプレットのアイコンが選択されるときに表示されるメニュー147Aiのオプションとしてユーザに利用可能にされる。例えば、ポートフォリオ・マネージャの方法146A1は、“Portfolio（ポートフォリオ）”メニュー147A1上にオプションとして表示される。通常のGUI方式では、方法／オプションの一つがそのペアレント・メニューから続いて選択されるとき、そのオプションのサブメニュー、またはページは、次いでJWS150Aによって表示されかつユーザ対話に対してイネーブルされる。サブメニュー147Aijの多くは、好ましい実施例によって供給される；例えば、Project->Create, Import, Choose, 及びRemoveサブメニュー147A1a、147A1b、147A1c、147A1d及びProject->Create, Import, Choose, Remove, Run及びCopyサブメニュー147A2a、147A2b、147A2

15

c、147A2d、147A2e、147A2fは、サブメニュー関連方法を適用することができるポートフォリオ及びプロジェクトのリストをそれぞれ供給する。例えば、図4を参照すると、Portfolio->Choose 方法146A1cに対する一組のポートフォリオ(Portfolio1、Portfolio2、Portfolio3)をリストしているPortfolio->Choose サブメニュー147A1cの一例が示されている。図4は、また、ポートフォリオ・マネージャ方法146A1(Create, Import, Choose, Remove)をリストしているポートフォリオ・メニュー147A1を示す。

【0033】図3を再び参照すると、JWSアプレット140Aiの方法146Aiは、それらの対応するメニュー147Aiサブメニュー147Aijを扱っているユーザの視点からここで説明される。この説明のほとんどは、JWS150Aのキー構成要素である、ポートフォリオ及びプロジェクト・マネージャ・アプレット114A1、114A2の方法に集中する。

【0034】ポートフォリオ・マネージャ方法ポートフォリオ・マネージャ・アプレット140A1は、JWS150Aのユーザに“Create(生成)”146A1a、“Import(取り入れ)”146A1b、“Choose(選択)”146A1cかつ“Remove(除去)”146A1dポートフォリオをそれぞれ許容させる4つの方法146A1を供給する。これらの方法146A1の各々は、それぞれが局所または遠隔システムに記憶されるしかつ一つのポートフォリオを表す、一組のポートフォリオ・ファイル160Aiと対話することによってその対応タスクを達成する。図3に示すように、一般的なポートフォリオ・ファイル160Aは、そのポートフォリオの成分プロジェクト・ファイル170Aに対する参照164A1iのコレクションを含む。本発明における他のファイル参照では、プロジェクト参照162Aiは、参照が局所ファイル名(“Name”(名前))であるような場合の、局所的に記憶されたプロジェクトに対するもの、または参照がURLであるような場合の、ウェブ文書に対するものでありうる。

【0035】例えば、図5を参照すると、その全てがメモリ106Aのユーザの“home(ホーム)”(即ち、局所)ディレクトリに記憶された局所プロジェクトである、“Applet(アプレット)”、“Standalone(スタンドアロン)”プログラム、Java“Package(パッケージ)”、“Image(イメージ)”及び“Remote(リモート)”アプレットを含む、その成分プロジェクトに対するプロジェクト・ファイル参照162A1jを含むポートフォリオ・ファイル160A1が示されている。これらのプロジェクトがユーザの“home(ホーム)”ディレクトリに全てが記憶されるので、それらは、ユーザによって読み取りかつ書き込むことができかつそれらの対応プロジェクト・ファイルは、経路(パス)及びファイ

16

ル名によって参照することができる。例えば、アプレット・プロジェクト・ファイル170A1への参照162A1aは、“/home/Applet.prj”である。ポートフォリオ・ファイル160A1は、また、マシンAのライブラリ・ディレクトリに記憶された読取り専用プロジェクト(“SemiRemote”)に対するプロジェクト・ファイルへの参照162A1f(/lib/SemiRemote.pri)、及びJWSブラウザ154Aを用いてインターネット上でのみアクセスすることができるマシンBに記憶された読取り専用プロジェクト(“Internet”)に対するプロジェクト・ファイル170B1への参照162A1g(/http://B.com/Internet.pri)参照を含む。

【0036】図3を再び参照すると、好ましい実施例では、各ユーザは、そのユーザに属するプロジェクトだけを含む(対応しているポートフォリオ・ファイル160Aiを有する)パーソナル・ポートフォリオを有する。JWS150Aが最初に起動されるとき、それは、現行、またはアクティブ、ポートフォリオとしてパーソナル・ポートフォリオを送り出す(bring up)。ポートフォリオ・マネージャの“Choose”方法/オプション146A1cを用いて、ユーザは、現行ポートフォリオであるべく別のポートフォリオ160Aiをchoose(選択すること)ができる。ユーザは、所望のポートフォリオのファイル名(もしそれがローカルならば)またはURL(もしそれが遠隔ならば)を、全ての利用可能なポートフォリオをリストしているPortfolio->Choose サブメニュー147A1c(この用語は、Portfolio(ポートフォリオ)メニュー147A1からのChoose(選択)オプションのユーザの選択に続いてJWS150Aによって表示されるChoose(選択)サブメニューを指定する)から選択することによってこれを行う。ユーザは、次いで、JWSツールバー160からポートフォリオ・マネージャ・アイコン1A1を選択することによって現行ポートフォリオから構成されているプロジェクトを見ることができる。ここに記載されたこの及び他の状況において、実行されているアプレットは、アプレット・ウィンドウ164A上にその結果及びメニューを表示する。

【0037】JWS150Aのユーザは、ポートフォリオ・マネージャの“Create”オプションを選択し次いで生成されるべきポートフォリオの名前を入力することによって新しいポートフォリオをcreate(生成)することができる。それに応じて、JWS150Aは、局所システム上に対応しているポートフォリオ・ファイル160Aを生成し、ツールバー160Aにその名前を表示しかつChoose及びRemoveサブメニュー147A1c、147A1dにポートフォリオの名前を加える、Portfolio->Create 方法146A1aを呼出す。新しく生成されたポートフォリオは、プロジェクトを有していないが、ユーザは、Portfolio->Create サブメニュー147A2a(以下に説明する)にプロジェクトを加えるかまたは既

存のプロジェクトを、Project->Import メニュー項目 147A2b (これも以下に説明する) を有するポートフォリオの中に取り入れることができる。一度新しいポートフォリオが生成されたならば、そのクリエイタは、それをプライベートに維持することができるかまたは他のものによってアクセスされるようにインターネット上にそれをパブリッシュ (発行) することができる。

【0038】ユーザは、また、それらのPortfolio->Choose サブメニュー147A1cには現在存在していない既存のポートフォリオをインポートすることもできる。10 そのようなポートフォリオをインポートするために、ユーザは、まず、Portfolio メニュー147A1にリストされたImportオプションを選択する。これは、ユーザがインポートされるべきポートフォリオのファイル名またはURLを入力するようなネーム・フィールドを有するインポート・サブメニュー、及び (それらがそれらの) 15 入力を終了したときにユーザがクリックスするインポート・ボタンを送り出すべくPortfolio->Import 方法146A1bをトリガする。それに応じて、インポート方法147A1cは、ポートフォリオ名をPortfolio->Choose 及びPortfolio->Remove サブメニュー147A1c、147A1dに加える。JWS150Aは、また、現行ポートフォリオをインポートされたポートフォリオに変更する。一度それがPortfolio->Choose サブメニュー147A1cの上にあるならば、インポートされたポートフォリオは、他のポートフォリオのように扱うことができる。

【0039】ユーザは、Portfolio->Remove サブメニューから除去されるべきポートフォリオを選択することによってポートフォリオをremove (除去) することができる。それに応じて、JWS154Aは、Choose及びRemoveサブメニュー147A1c、147A1dから選択されたポートフォリオを除去する、Portfolio->Remove方法146A1dを呼出すが、ポートフォリオの対応するポートフォリオ・ファイル160Aを削除しない。ポートフォリオ・ファイルがユーザのシステムから削除されないで、ユーザは、Portfolio->Import オプション146A1bを用いてポートフォリオをいつでもインポートすることができる。ポートフォリオに含まれた各プロジェクト・ファイルは、プロジェクトを記述しかつプロジェクトの内容を含む対応プロジェクト・ファイル170Aを有する。より特定のには、各プロジェクト・ファイル170Aは、次の情報を含む：

【0040】(1) プロジェクトの名前172A；
(2) プロジェクト型174A (Javaアプレット (APPLET)、スタンドアロン・プログラム (STANDALON)、Javaクラス・ライブラリ (PACKAGE)、データ・ファイル (IMAGE)、これら4つの先に示した型の一つの遠隔プロジェクトのインポートされたコピー、または遠隔アプレット (REMOTE))；

(3) プロジェクトに対するソース・コード178Aをインターネット上でプロジェクトを要求している他のもの及びプロジェクト・オプション180Aに分散すべきか否かを含んでいる、プロジェクト管理情報176A；
(4) 実際のプロジェクト内容及び/又は他のプロジェクト・ファイル170Aiに対する一組の参照を含むことができ、埋込まれたプロジェクトの多重レベルをイネーブルしている、プロジェクト内容182A；及び
(5) アプレット・プロジェクトに対するアプレット・タグを含むHTMLファイルのURLである、ラン・ページ (run page) URL184A (アプレット・プロジェクトに対してのみ適用可能)。

【0041】この情報は、JWS150Aによって供給されたプロジェクト方法のどれが特定のプロジェクト上のユーザによって採り入れることができるかを決定する。これらのプロジェクト方法146A2をここで説明する。

【0042】プロジェクト方法

JWS150Aは、プロジェクトを扱うための複数の方法を供給する。これらの方法は、Project (プロジェクト) メニュー147A2上のオプションとしてユーザに利用可能である。これらの方法/オプションの一つが選択されるとき、JWS150Aは、(それから) ユーザがさらなる操作の詳細を特定する対応サブメニュー147A2jを表示する。プロジェクト方法146A2は、: Create (クリエート) 146A2a、Import (インポート) 146A2b、Choose (チョーズ) 146A2c、Edit (エディット) 146A2d、Remove (リムーブ) 146A2e、Run (ラン) 146A2f、Copy (コピー) 146A2g、及びPaste (ペースト) 146A2h。

【0043】これらの方法は、ユーザに、既存のプロジェクト (局所または遠隔) を扱わせるか新しいプロジェクトを生成させる。いずれの場合においても、プロジェクトは、ポートフォリオのコンテキストに常に存在する。プロジェクトが生成されるとき、それは、現行ポートフォリオにおける現行プロジェクトになる。ユーザは、Javaアプレット・プロジェクト、スタンドアロン・プロジェクト・プロジェクト、Javaパッケージ・プロジェクト、イメージ・プロジェクトまたは遠隔プロジェクトをcreate (生成) することができる。これらのプロジェクトのいずれかを生成するために、ユーザは、まず、プロジェクトがそれに関連付けられるべきポートフォリオを "Chooses" (選択し) かつProject (プロジェクト) メニュー147A2から "Create (生成)" オプションを選択して、それにより、JWS150Aは、Project->Create 方法146A2aを呼出す。この方法146A2aは、その上でユーザが (彼らが) 生成したいプロジェクトの型を選択する、Project->Create ページ147A2aを表示する (例えば、ユーザが

アプレットを生成したいならば、(彼らは、)サブメニュー上に表示されたアプレット・ボタンの上をクリックする)。ユーザは、次いで、生成されるべきパッケージの名前、及びパッケージの対応しているプロジェクト・ファイル170Aiがその中に記憶されるべきメモリ106Aの局所ディレクトリを特定する。一度ユーザがプロジェクトに対する属性を特定したならばProject->Create方法は、プロジェクトの対応しているプロジェクト・ファイルに対する参照162Aijを現行ポートフォリオ160Aiのポートフォリオ・ファイル170Ai 10に加える。

【0044】ある情況(生成されているプロジェクトがアプレット、スタンドアロン、プロジェクトまたはJavaパッケージであるときには)では、ユーザは、また、新しく生成されたプロジェクトに対するソース・コードへのアクセスを有しうる。これらの情況では、ユーザは、Project->Create ページ147A2a上の対応しているソース・ファイルのファイル名を入力する。JWS150Aは、これらのソース・ファイル名を、ソース・ファイルがユーザによってアクセスできるようにメモリ106Aに維持(保守)された“Sources”(ソース)リストに加える(例えば、編集及びコンパイルのための)。ユーザは、また、新しく生成されたプロジェクトがその一部であるプログラムに対する主ファイル(即ち、“主”ルーチンを含むファイル)の名前を入力する。生成されているプロジェクトがJavaアプレットであるとき、アプレットの参照が選択されるときに、アプレットが実行されるように、JavaアプレットがHTMLページにおいて参照されるということが可能である。JWS150Aは、ユーザがアプレットを実行するHTMLページの名前をオプション的に入力するようなProject->Create(プロジェクト生成)ページにおけるRun Page(ラン・ページ)URLフィールドを介してあらわされるべくそのような関係を許容する。

【0045】ユーザがイメージ・プロジェクトを生成しているとき、Project->Create サブメニュー147A2aから“image”(イメージ)をChoosing(選択した)後、ユーザは、イメージ・プロジェクトの名前及び対応するイメージ・ファイルのURLを入力する。ユーザは、次いで、:

【0046】(1) 周囲テキストに関するイメージの位置合わせ(例えば、“bottom”(底部)位置合わせをchoosing(選択すること)は、イメージを表示しているブラウザに、テキストの底部にイメージの底部を位置合わせさせる);

(2) イメージがアクティブであるか否か(イメージを見ている人が、異なる動作を生成するためにイメージの異なる領域上でクリックすることができることを意味する);

(3) イメージを表示することができないブラウザに 50

よってイメージの代わりに表示することができるオプション的テキスト・ストリングのような、イメージに関連付けられた属性をオプション的に入力することができる。

【0047】一度ユーザが、彼らが生成しているイメージ・プロジェクトに対してこの情報を記入したならば、ユーザは、Project->Create ページ147A2aの“Apply”(アプライ)フィールド上でクリックし、それによりProject->Create 方法146A2aは、新しく生成されたイメージ・プロジェクトを現行プロジェクトにしてかApplet(アプレット)ウィンドウ164Aにイメージを表示する。Create(生成)方法146A2aは、また、Project(プロジェクト)メニュー147A2のChoose(チューズ)、Edit(エディット)及びRemove(リムーブ)サブメニュー147A2c、147A2d、147A2eにイメージ・プロジェクト名を加えかつ対応するプロジェクト・ファイル170Aiの名前を現行ポートフォリオに関連付けられたポートフォリオ・ファイル160Aiに加える。

【0048】JWS150Aのユーザは、それらのパーソナル・ポートフォリオの一つの中にあらゆる型のプロジェクトをimport(インポート)することができる。彼らは、彼らが現行ポートフォリオであることを欲するポートフォリオを選択し、Project メニュー147A2から“Import”オプションを選択し、そしてProject->Import 方法146A2bによって表示されるProject->Import ページ147A2b上にインポートされるべきプロジェクトの名前またはURLを入力することによってこれを行う。必要な情報を入力した後、ユーザは、インポート・ページ147A2b上に表示された“import”ボタンをクリックし、それにより、インポート方法が146A2bが指定されたプロジェクトを現行ポートフォリオの中にインポートしかつプロジェクト名/URLをProject->Choose、Edit、Remove、及びRun サブメニュー147A2c、147A2d、147A2e及び147A2fに加える。インポート方法146A2bは、また、それが既にその中に含まれていないならば、インポートされたプロジェクトのプロジェクト・ファイル170Aiの名前を現行ポートフォリオに加える。JWS150Aは、インポートされたプロジェクトを現行プロジェクトにしないが、しかし、ユーザがツールバー160からPortfolio Manager(ポートフォリオ・マネージャ)アイコンIA1を続いて選択するならば、JWS150Aは、インポートされたプロジェクトを表示する。

【0049】JWS150Aは、ユーザに遠隔アプレット(a remote applet)プロジェクトを生成(create)させる。ユーザは、現行ポートフォリオを“Choosing(選択)”し、“Project(プロジェクト)”メニューから“Create(生成)”オプションを選択(select)しかつProject->Create サブメニュー147A2a上に表示され

た“遠隔アプレット(remote applet)” ボタンをクリックすることによってこれを行う。ユーザは、次いでプロジェクトの名前及びアプレットを実行するHTMLページのURLを入力する。一度これらのフィールドが終了されると、ユーザは、“Apply (アプライ)” 上でクリックすることによりProject->Create サブメニュー147A2aを出る。生成方法(create method) 146A2は、次いで空内容182Aiを有するプロジェクト・ファイル170Aiと、アプレットを実行するHTMLページのURLに設定されるラン・ページURLフィールド184Aiを生成する。例えば、図5を参照すると、遠隔プロジェクト・ファイル170A5は、遠隔アプレット“Applet2” を走らせるウェブ・ページ(“RunApplet2.htm”) のURL (“http://C.com/RunApplet2.htm”) に設定されたラン・ページURL 184A5を有する。生成方法146A2は、また、プロジェクト・ファイル170Aiの名前を現行ポートフォリオのポートフォリオ・ファイル160Aiに加える。JWS150Aは、次いで、インポートしたプロジェクトを現行プロジェクトにし、Portfolio Manager (ポートフォリオ・マネージャ) 140A1をロードしかつPortfolio Manager 140A1によって表示されるべき現行プロジェクトを選択する。JWS150Aは、次いでインポートされたアプレット・プロジェクトの名前をProject (プロジェクト) メニュー147A2のChoose (選択), Edit (編集), Remove (除去), Run (ラン) 及びCopy (コピー) サブメニュー147A2c、147A2d、147A2e、147A2fに加える。

【0050】ユーザは、次いで、Project->Runサブメニュー147A2fからその名前を選択するかまたはPortfolio Manager (ポートフォリオ・マネージャ) をロードし、遠隔プロジェクトを選択し、そしてツールバー160上のRun (ラン) ボタン1A7を押下げることによって遠隔アプレットを走らせることができる。Project->Run方法146A2fは、次いで、参照されたウェブ・ページ (http://C.com/RunApplet2.htm) をダウンロードしかつ遠隔アプレット (Applet2) を走らせる、ウェブ・ブラウザ(web browser) 154Aに、遠隔アプレット・プロジェクト・ファイル (170A5) のラン・ページURLフィールド (例えば、184A5) において参照されたウェブ・ページのURLをパスする。ユーザがアプレットのプロジェクト・ファイル170Aiにおいてラン・ページ (Run page) URL 184Aiを特定しないならば、そのアプレット・プロジェクトは、Project->Run方法146A2fを用いてまだ走らせることができる。この状況では、Project->Run方法146A2fは、Edit Project (編集プロジェクト) 146A2dラン・フォルダー(run folder) にユーザによって入力されたプロジェクト属性及びパラメータで生成されたアプレット・タグを含む新しいウェブ・ページ (web page) を

自動的に生成する。この自動的に生成されたHTMLページは、アプレット・プロジェクトを走らせるためにブラウザ154Aを用いる、JWS150Aの中にロードされる。この特徴は、参照しているアプレットに対するHTML構文を知る必要なくユーザにアプレットを実行させる。

【0051】プロジェクト・マネージャ (Project Manager) 140A2のコピー方法 (Copy method) 146A2gは、JWS150Aのユーザに、ユーザに参照しているアプレットに対するHTML構文を知ることを要求せずにアプレットを実行するHTMLファイルの中にアプレットをコピー (copy) させる。ユーザは、現行ポートフォリオにおけるアプレット・プロジェクトを最初に選択し (その上で単一クリックし) そしてプロジェクト (Project) メニュー147A2からCopy (コピー) オプションを選択することによってこれを行う。このアクションのセットは、コピー方法 (Copy method) 146A2gに、選択されたアプレット・プロジェクトの内容182AiをJWS150Aによって維持されるクリップボード (図示省略) にコピーさせる。ユーザは、次いで、ツールバー160Aからテキスト・エディタ・アイコン (Text Editor Icon) 1A3を選択し、それによりJWS150Aは、エディタ方法 (Editor method) 146A2dを実行する。エディタ方法146A2dは、“Paste (ペースト)” を含んでいる、編集オプションのリストを含む、編集メニュー (Edit menu) 147A2dを含んでいるテキスト・エディタを送り出す。ユーザは、編集メニュー (Edit menu) 147A2dから“Paste (ペースト)” オプションを選択し、それにより、ペースト方法142A2hは、クリップボードの内容 (即ち、コピーされるアプレット) を新しいファイルの中にペーストする。ユーザは、次いで、セーブされたHTMLファイルにコピーされたアプレットへの適切なリンクをJWS150Aに加えさせる、HTMLファイルとして新しいファイルをセーブすることができる。他の新しいプロジェクトのように、JWS150Aは、新しいHTMLファイル170Aiのファイル名を現行ポートフォリオのポートフォリオ・ファイル160Aiに加える。代替的に、ユーザは、彼らがアプレットを含むことを欲するHTMLファイルのイメージ上に、コピーされるべきアプレットのイメージを単にドラッグすることができる。JWS150Aは、次いで、アプレットの内容を指定されたHTMLファイルにコピーしかつコピーされたアプレットを参照しているHTMLファイル・タグに加える。

【0052】Project->Edit 方法146A2dは、また、ユーザに、全ての型のプロジェクトを編集 (edit) させる。編集方法 (Edit method) 146A2dは、二つの方法の一つでJWS150Aのユーザによって呼出すことができる。第1に、ユーザは、現行プロジェクト上のエディティング (即ち、エディティング方法146A2

d)を呼出すべくツールバー160上に表示された編集プロジェクト・アイコン(Edit Project icon)1A3上でクリックすることができる。第2に、ユーザは、Project->Edit サブメニュー147A2dから編集されるべきプロジェクトの名前を選択することができる。一度エディティングが指定されたプロジェクトに対して選択されたならば、JWS編集方法(Edit method)146A2dは、ユーザが指定されたプロジェクトに対する情報を編集することができるような6つのフォルダーを含む編集ページ(edit page)147A2dをオープンする。これら6つのフォルダー及びそれらの関連情報は、:

【0053】General 名前、型及びソース・ディレクトリを含んでいる、プロジェクトについての情報
Build プロジェクトをコンパイルするために必要な情報
Debug/Browse ソース・ファイルをデバッグしかつブラウズするために必要な情報
Run JWSブラウザにおけるアプレットまたはスタンドアロン・プログラムを実行するために必要な情報;
Publish プロジェクトを他のユーザによってコピーさせるために必要な情報
Portfolio ポートフォリオ・マネージャ(Portfolio Manager)においてプロジェクトを表示するために必要なポートフォリオ(Portfolio) 情報を含む。

【0054】Project->Edit 方法146A2dは、適切なところにおいてのみ、ユーザに、これら6つのフォルダーにおけるフィールドを編集させる。ユーザを支援するために、編集方法146A2dは、不適用(inapplicable)フィールドをグレイ・アウト(grey out)する。フィールドが適用可能であるか否かは、編集されるプロジェクトの型及びプロジェクトが局所であるかまたは遠隔であるかに依存する。例えば、編集方法146A2dは、ユーザに、ソース・ファイルではないプロジェクトに対するDebug/Browse(デバッグ/ブラウズ)フォルダーにおけるフィールドを編集させない。General, Build, Debug/Browse及びRun フォルダーにユーザによって入力することができる情報は、ほとんど慣習的であり、それゆえに、より詳細には説明しない。しかしながら、これらのフォルダーにおけるエディティング情報について固有なことは、JWS150Aがユーザに、ファイル名またはURLsによって識別された遠隔と共に局所プロジェクトに対する情報を供給させるということである。これは、ユーザに、例えば、デバッグまたはブラウズされるべき特定のプロジェクトに対するソース・コードがある遠隔ノード上に存在するということを特定させる。これは、通常のプロジェクト及びファイル管理システムでは可能ではない。

【0055】JWS150Aの好ましい実施例がユーザ

に、遠隔ソースからポートフォリオ及びプロジェクトを採り入れさせかつ他のものの使用に対してそれら自体のポートフォリオ及びプロジェクトを発行させるので、この実施例は、また、インターネット上のプロジェクトの発行に関連するプロジェクトのある一定の属性を示すべくプロジェクトのクリエータのための方法を供給する。これらの発行属性(publication attributes)は、次に示すフィールドを含む、Public及びPortfolio Folders に含まれる:

10 【0056】Portfolio Folder

Description マウスがポートフォリオ・マネージャにおいてプロジェクト・イメージ上に配置されるときにJWSブラウザによって表示されるプロジェクトの簡単な説明;

Portfolio image URL

ポートフォリオにおけるプロジェクト・イメージを表すイメージ・ファイル(GIF, JPEG, または他のもの)に対するURL(イメージ・ファイルが特性されれば、デフォルトGIFファイルがJWS150Aによって用いられる);及び

20 Features プロジェクトの一般特性、例えば、プロジェクトがビデオ、グラフィックスまたは音である。

【0057】Publish Folder

Distribute source copies

プロジェクトが一つのポートフォリオから別のものにコピーされるときにプロジェクトのソース・ファイルがコピーされるか否かを制御する二つの値(はい/いいえ)を有するトグル・フィールド(このフィールドが“いいえ”に設定されるときに、JWSは、対応するプロジェクト・ファイル170Aiに内容182Aがアブセント(無し)でコピーさせるだけである);及び

Submitter Name, E-Mail及びURL

プロジェクトをポートフォリオに追加している人の名前、電子メール・アドレス及びウェブ・ページ。

【0058】ユーザは、二つの方法の一つで現行プロジェクト(即ち、JWS150Aにおいて作動されているプロジェクト)を変更することができる。第1では、ユーザは、JWSツールバー160からポートフォリオ・マネージャ(Portfolio Manager)アイコン1A1を選択することによって開始する。これは、JWS150Aに、アプレット・ウィンドウ(Applet Window)164Aにおいて現行ポートフォリオのプロジェクトを示しているポートフォリオ(Portfolio) ディスプレイをオープンさせる。ユーザは、次いで、彼らが現行プロジェクトであることを欲するプロジェクトをポートフォリオ(Portfolio) ディスプレイから選択する。JWS150Aは、選択されたプロジェクトを現行プロジェクトにしかつ現行プロジェクトの名前をJWSツールバー160上に表示する。代替的に、ユーザは、Project->Choose サブメニュー147A2cからプロジェクト名を選択する(cho

osing)ことによって現行プロジェクトを変更することができる。

【0059】JWS150Aは、次に示す方法の一つでユーザにポートフォリオからプロジェクトを除去(remove)させる。第1に、アプレット・ウィンドウ164Aのポートフォリオ・マネージャ・ディスプレイでは、ユーザは、彼らが除去することを欲するプロジェクトを選択しそしてポートフォリオ・マネージャ(Portfolio Manager)140A1によって供給されたRemove(除去)アイコン(図示省略)をクリックすることができる。代替的に、彼らは、Project→Removeサブメニュー147A2eから除去されるべきプロジェクトの名前を選択する(choose)ことができる。いずれの場合においても、一度ユーザが除去されるべきプロジェクトを示したならば、Project→Remove方法146A2eは、プロジェクト・メニュー(Project Menu)147A2のChoose, Edit, Remove及びCopyサブメニュー147A2c、147A2d、147A2e、147A2gからプロジェクトを除去する。Project→Remove方法146A2eは、除去されたプロジェクトのプロジェクト・ファイル170Aiを削除しないということに注目する。これは、ユーザがもし必要ならば(Project→Import 方法146A2bを用いて)後でプロジェクトを続いてインポートすることができることを確実にする。

【0060】図6を参照すると、本発明の合成利点を強調するJWS150Aのユーザ・インターフェイスの説明を示す。JWS150Aの8つのアイコンIA1～IA8がツールバー160上に示されている。二つのポートフォリオ(ポートフォリオ(Portfolio)1及び(Portfolio)ポートフォリオ2)が、ポートフォリオ・マネージャ140A1の制御下である、アプレット画面164A上に示されている。これらのポートフォリオは、JWS150Aに収容することができる異なる型のポートフォリオの二つを表す。ポートフォリオ1は、対応しているプロジェクト・ファイル170A1a、170A1b、170A1cを有する3つの局所プロジェクトP1a、P1b、P1cを含むポートフォリオ・ファイル160A1を有する局所ポートフォリオである。ポートフォリオ1の斜線で示されたプロジェクトP2a'は、JWS150Aの制御下のJWSブラウザ154A(図1)によりインターネットにわたりポートフォリオ2からインポートされた遠隔ポートフォリオである。このプロジェクトP2a'は、プロジェクト・ファイル170A2a'によりメモリ106Aに表される。ポートフォリオ1がミックスされても、そのプロジェクト全ては、JWS150Aにおいて同じファッショで操作できる。

【0061】ポートフォリオ2は、そのコンポーネントもまた遠隔的に記憶される遠隔ポートフォリオである。JWS150Aは、JWSブラウザ154Aを用いてイ

ンターネットにわたりポートフォリオ2のコンポーネントをアクセスするが、ポートフォリオ1と同じ方法でポートフォリオ2を表示する。ポートフォリオ1のような局所ポートフォリオもまた遠隔的に記憶されたプロジェクト(例えば、マシンBに記憶されたプロジェクト)だけを参照するというに注目する。インターネットにわたり分散されうるプロジェクト及びポートフォリオを編成すべき一つの継目もないJWS150Aのこの機能は、JWS150A及びJWSウェブ・ブラウザ154Aの統合による。本発明のインターネット認識は、また、それらがインターネット上で他のものによってアクセスされかつ用いられることができるように、ユーザに彼ら自身のポートフォリオを発行させることができる。

【0062】本発明は、2～3の特定な実施例を参照して記述されたが、記述は、本発明の説明のためであり本発明を限定することを意図していない。種々の変更が、特許請求の範囲によって画定された本発明の真の精神及び範疇から逸脱することなく当業者において生じるであろう。

【0063】

【発明の効果】本発明のシステムは、ディスプレイ及びメモリを有している第1のコンピュータにおいて、第1のコンピュータは、一組の遠隔コンピュータでネットワークされており、そのコンポーネントがコンピュータ上でだけ存在すべく拘束されていない複合文書のローディング及び実行を第1のコンピュータ上に表示された図的ユーザ・インターフェイスから起動するためのシステムであって：一組の選択可能なアイコンを有しており、各アイコンが当該アイコンの選択を介して起動される一つ以上の操作に関連付けられている、表示可能ツールバー；それぞれが関連付けられた操作の一つ以上を実行するために必要なコンポーネント、またはファイルに対する一組の参照を含む、一組の複合文書；アプレットまたはデータ・ファイルから選択されたファイル型及び局所または遠隔から選択された位置を有しているファイルのサブセットのそれぞれ；それぞれがアイコンの一つを文書の組の一つ以上に関連付け、かつそれぞれが文書がコンピュータ上に記憶されるときに局所リンクからまたは文書が遠隔コンピュータ上に記憶されるときにネットワーク・リンクから選択される、一組のリンク；及びアイコンの一つが選択されるときに、コンピュータに、一組のリンクを介して選択されたアイコンに関連付けられた文書において参照されたファイルをロードさせかつ実行可能であるロードされたファイルのいずれかを実行させるように構成され、それによってアイコンに関連付けられた操作を起動する、ブラウザを備えているので、局所または遠隔ファイル及びプログラムに関連付けることができるアイコンを供給することができる。

【0064】本発明の方法は、コンポーネントのサブセットをホストすることができる一組の遠隔コンピュータ

とネットワークされている第1のコンピュータ上にだけその該コンポーネントが存在すべく拘束されていない複合文書のローディング及び実行を第1のコンピュータ上に表示された図的ユーザ・インターフェイスから起動する方法であって：一組の選択可能なアイコンを有しているツールバーを表示し、各アイコンは、当該アイコンの選択を介して起動される一つ以上の操作に関連付けられかつ一つ以上の複合文書にリンクされ、それぞれは、関連操作の一つ以上を実行するために必要なコンポーネント、またはファイルに対する一組の参照を含み、ファイルのサブセットのそれぞれは、アプレットまたはデータ・ファイルから選択されたファイル型を有しており；アイコンの一つが選択されることにより、選択されたアイコンにリンクされた一つ以上の複合文書をローディングし；一つ以上の複合文章をローディングすることにより、複合文書で参照されたファイルをローディングし、ファイルは、第1のコンピュータ上にだけ存在すべく拘束されておらず；かつロードされたファイルがアプレットまたはスタンドアロン・アプリケーションから選択される型を有するとき、ロードされたファイルを実行する段階を具備するので、局所または遠隔ファイル及びプログラムに関連付けることができるアイコンを供給することができる。

【図面の簡単な説明】

【図1】ネットワークされたコンピュータの一つに関連付けられたメモリ及びディスクの詳細を表しているコンピュータ・ネットワークのブロック図である。

【図2】ツールバーからのアイコンの選択に続いて本発明によって実行される処理段階を示しているデータ・フロー図である。

【図3】Java Workshopの好ましい実施例に採り入れられたデータ構造を示している図である。

【図4】ポートフォリオ・マネージャ・メニュー及びボ

ートフォリオ・マネージャ方法の一つに関連付けられるサブメニューの例を示す。

【図5】Java Workshopの好ましい実施例に採り入れられるポートフォリオ・ファイルの構造を示す。

【図6】表示されたコンポーネントのあるものがポートフォリオ・マネージャ・ウィンドウを表示しているコンピュータに対して局所または遠隔であるような本発明のポートフォリオ・マネージャによって生成されたディスク・ウィンドウの図である。

【符号の説明】

100 コンピュータ・ネットワーク

102 コンピュータ

104 プロセッサ

106 メモリ

108 ディスプレイ

110A JWSファイル

112A JWSツールバー仕様

114A アイコン仕様

116A1、A2 アイコンの視覚的属性

118A1、A2 リンク

120A ウェブ文書

122A1、A2 タイトル

124A1、A2 参照

140A JWSアプレット

148A 参照ファイル

150A Java Workshop (JWS) プログラム

154A 統合型JWSブラウザ

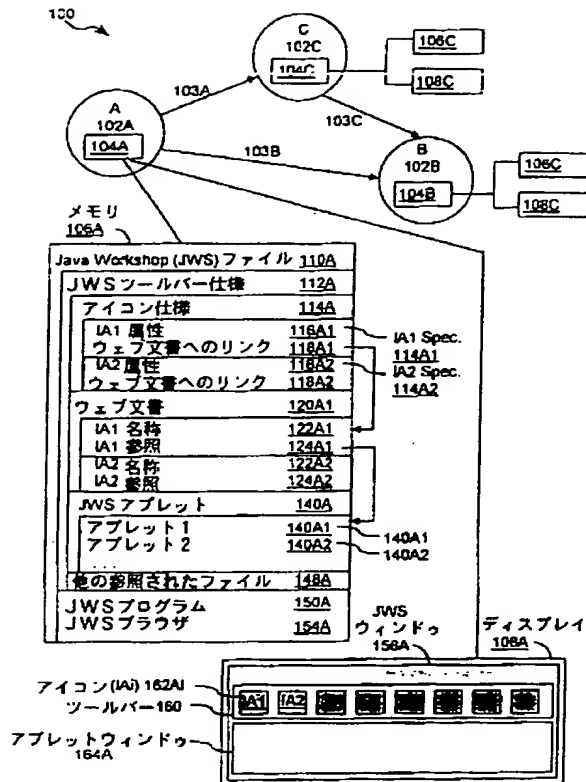
156A JWSウィンドウ

160 JWSツールバー

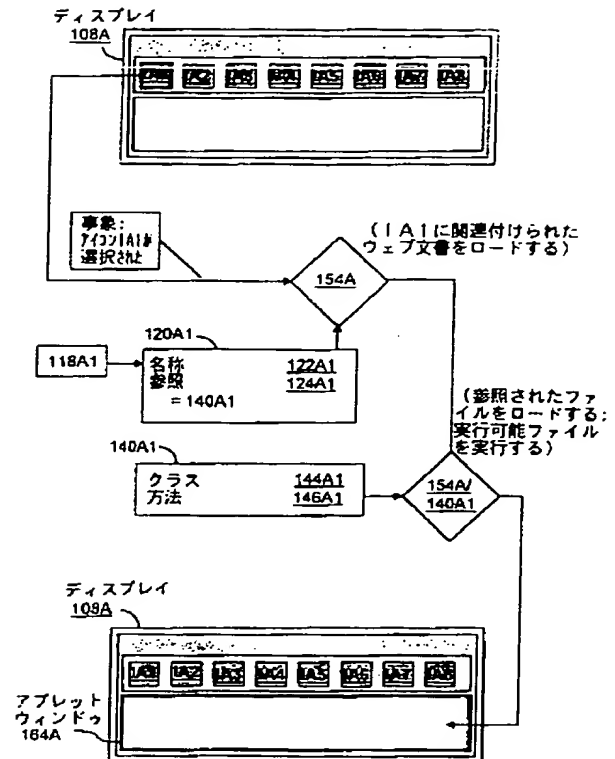
162Ai アイコン

164A アプレット・ウィンドウ

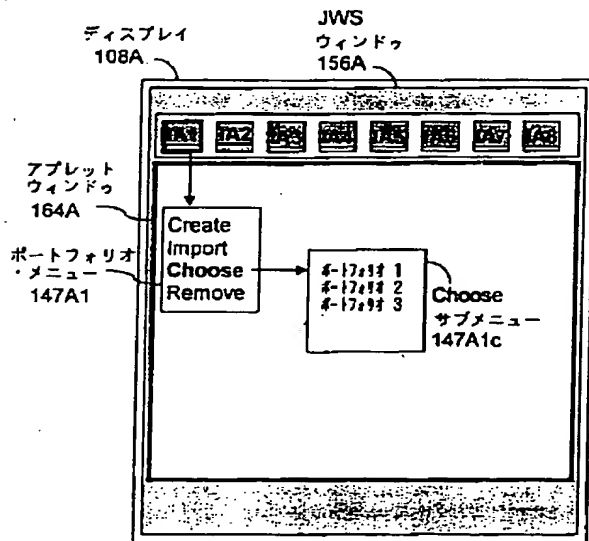
【図1】



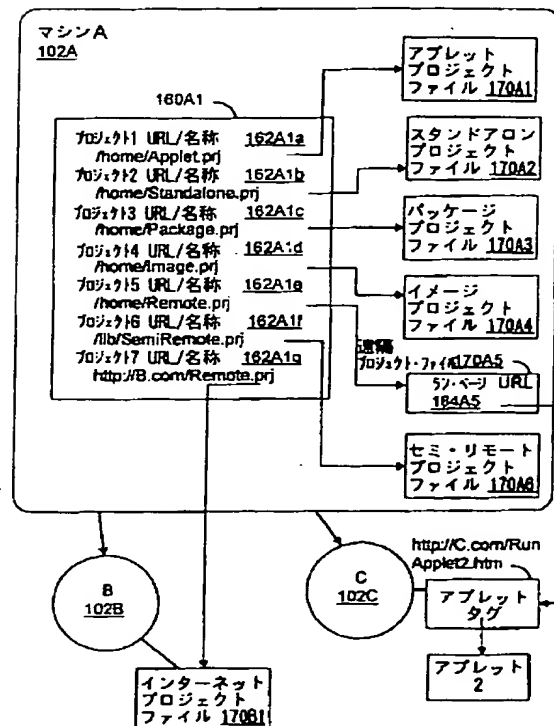
【図2】



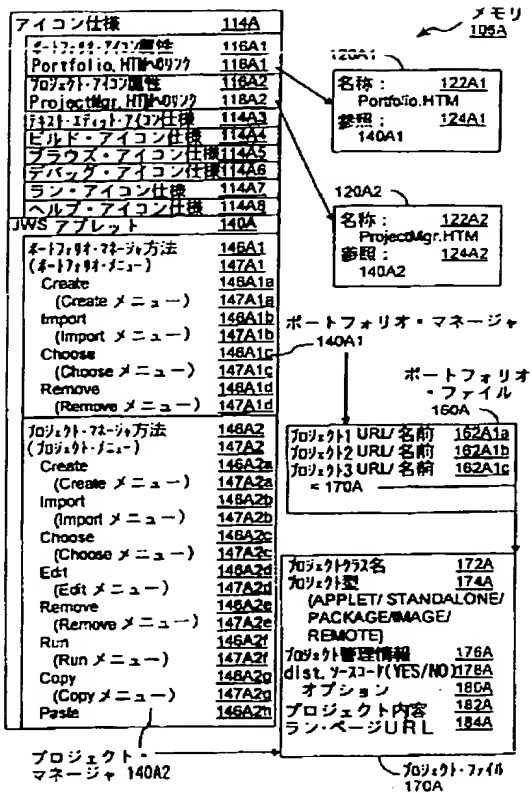
【図4】



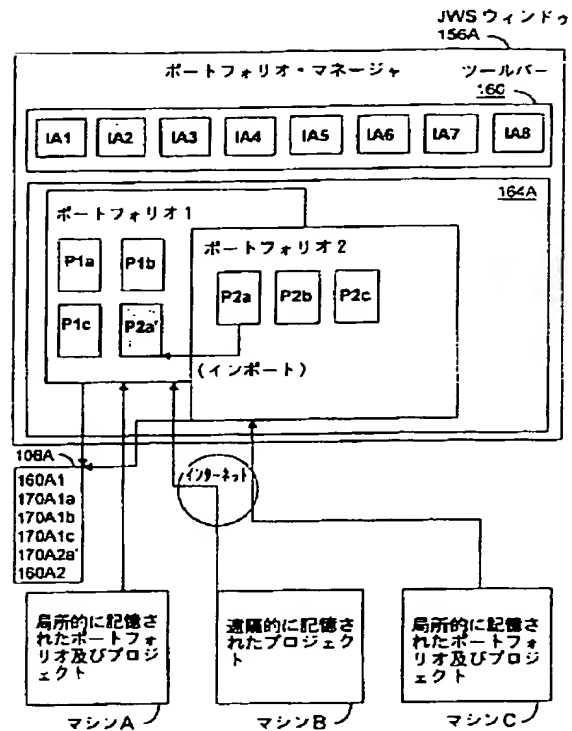
【図5】



【図3】



【図6】



フロントページの続き

(72)発明者 トーマス ディー ネヴィン
 アメリカ合衆国 カリフォルニア州
 94536 フリーモント ニコルス 36710

(72)発明者 ジル ポーラ フォーリー
 アメリカ合衆国 カリフォルニア州
 95117 サン ホセ ヴィクトリアン パ
 インズ 3708
 (72)発明者 カレン リン シエルスキー
 アメリカ合衆国 カリフォルニア州
 94087 サニーヴェイル ハイバーニア
 ウェイ 730